

# TD 1.1 – Initiation & Rappel

## A. Les commandes de base : Manipulation des fichiers et des répertoires

1. Quelle est la différence entre Unix et Linux ?

Unix est un SE alors que Linux est un noyau de SE

Unix est propriétaire, Linux est libre

2. Lancer votre terminal et afficher le contenu des variables d'environnement **PATH** et **SHELL**.

Expliquer.

La commande «**env**» affiche la liste des variables d'environnement.

```
meriam@meriamVM:~$ env
SHELL=/bin/bash
SESSION_MANAGER=local/meriamVM:@/tmp/.ICE-unix/1896,unix/meriamVM:/tmp/.ICE-unix/1896
QT_ACCESSIBILITY=1
COLORTERM=truecolor
XDG_CONFIG_DIRS=/etc/xdg/xdg-ubuntu:/etc/xdg
SSH_AGENT_LAUNCHER=gnome-keyring
XDG_MENU_PREFIX=gnome-
GNOME_DESKTOP_SESSION_ID=this-is-deprecated
LANGUAGE=en_US:en
LC_ADDRESS=fr_FR.UTF-8
GNOME_SHELL_SESSION_MODE=ubuntu
LC_NAME=fr_FR.UTF-8
SSH_AUTH_SOCK=/run/user/1000/keyring/ssh
XMODIFIERS=@im=ibus
DESKTOP_SESSION=ubuntu
LC_MONETARY=fr_FR.UTF-8
GTK_MODULES=gail:atk-bridge
PWD=/home/meriam
LOGNAME=meriam
XDG_SESSION_DESKTOP=ubuntu
XDG_SESSION_TYPE=wayland
SYSTEMD_EXEC_PID=1917
```

```
meriam@meriamVM:~/Desktop$ echo $PATH
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin
meriam@meriamVM:~/Desktop$ echo $SHELL
/bin/bash
```

La variable **PATH** est utilisée sur Linux pour contenir des chemins vers des programmes. Ces chemins permettent d'exécuter directement une commande dans le terminal.

La variable **SHELL** est utilisée pour indiquer l'interpréteur de commande par défaut de l'utilisateur.

3. Taper la commande **pwd**. Expliquer le résultat affiché ?

```
meriam@meriamVM:~/Desktop$ pwd
/home/meriam/Desktop
```

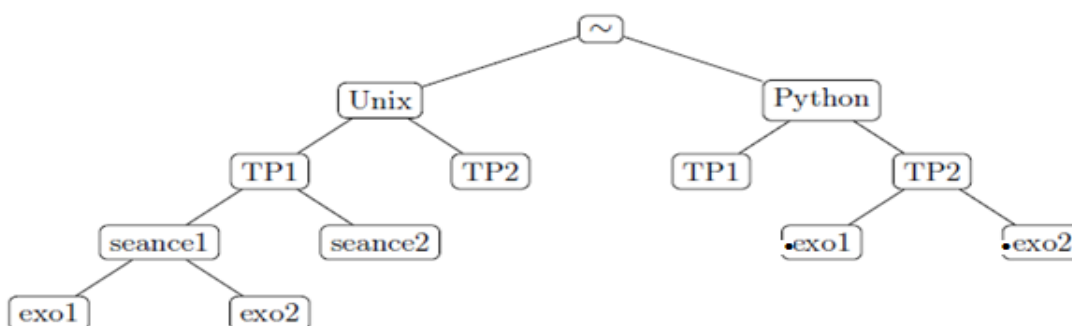
Cette commande affiche le chemin absolu de votre position actuelle à partir de la racine du système de fichiers.

4. Comment revenir dans votre répertoire personnel à partir de n'importe quel endroit dans le système de fichiers ?

On utilise l'une des commandes suivantes :

```
meriam@meriamVM:~/Desktop$ cd
meriam@meriamVM:~$
```

```
meriam@meriamVM:~/Desktop$ cd ~
meriam@meriamVM:~$
```



5. Soit l'arborescence ci-dessus. A partir de votre **répertoire personnel**, on veut créer cette arborescence. Pour ce faire :

- a. Créer deux répertoires « **Unix** » et « **Python** » dans votre répertoire personnel.

```
meriam@meriamVM:~/Desktop$ cd
meriam@meriamVM:~$ mkdir Unix Python
```

- b. Créer deux répertoires « **TP1** » et « **TP2** » dans le répertoire « **Unix** ».

```
meriam@meriamVM:~$ mkdir Unix/TP1 Unix/TP2
```

- c. Copier les deux répertoires « **TP1** » et « **TP2** » dans le répertoire « **Python** ».

```
meriam@meriamVM:~$ cp -r Unix/TP1 Unix/TP2 Python/
meriam@meriamVM:~$ ls Python/
TP1 TP2
```

- d. Créer deux répertoires « **seance1** » et « **seance2** » dans le répertoire « **TP1** » dans « **Unix** ».

```
meriam@meriamVM:~$ mkdir Unix/TP1/seance1 Unix/TP1/seance2
meriam@meriamVM:~$ ls Unix/TP1
seance1 seance2
```

- e. Créer deux fichiers « *exo1* » et « *exo2* » dans le répertoire « *seance1* ».

```
meriam@meriamVM:~$ touch Unix/TP1/seance1/exo1 Unix/TP1/seance1/exo2
meriam@meriamVM:~$ ls Unix/TP1/seance1/
exo1 exo2
```

- f. Copier les deux fichiers « *exo1* » et « *exo2* » dans le répertoire « *TP2* » de « *Python* » sous le nom de « *.exo1* » et « *.exo2* »

```
meriam@meriamVM:~$ cp Unix/TP1/seance1/exo1 Python/TP2/.exo1; cp Unix/TP1/seance1/exo2 Python/TP2/.exo2
meriam@meriamVM:~$ ls -a Python/TP2/
. . . .exo1 .exo2
```

6. A partir du répertoire « *Python/TP1* », essayer de se déplacer dans « *Unix/TP2* » en utilisant un chemin relatif.

```
meriam@meriamVM:~$ cd Python/TP1
meriam@meriamVM:~/Python/TP1$ cd ../../Unix/TP2
meriam@meriamVM:~/Unix/TP2$
```

7. Retourner au répertoire « *Python/TP1* » en utilisant le chemin absolu.

```
meriam@meriamVM:~/Unix/TP2$ cd /home/meriam/Python/TP1
meriam@meriamVM:~/Python/TP1$
```

8. Supprimer le répertoire « *TP1* » de « *Python* ».

```
meriam@meriamVM:~$ rmdir Python/TP1
meriam@meriamVM:~$ ls Python/
TP2
```

9. Supprimer le répertoire « *TP2* » de « *Python* » avec la commande *rmdir*, pourquoi la commande a-t-elle échoué ? Proposer une autre commande pour le supprimer.

```
meriam@meriamVM:~$ rmdir Python/TP2
rmdir: failed to remove 'Python/TP2': Directory not empty
```

La commande a échoué car le répertoire TP2 de Python n'est pas vide. On peut le supprimer comme suit :

```
meriam@meriamVM:~$ rm -r Python/TP2/
meriam@meriamVM:~$ ls Python/
meriam@meriamVM:~$
```

10. Déplacer « *exo1* » dans « *TP1* » et supprimer « *seance1* » et « *seance2* ».

```
meriam@meriamVM:~$ mv Unix/TP1/seance1/exo1 Unix/TP1/exo1; rm -r Unix/TP1/seance1 Unix/TP1/seance2
meriam@meriamVM:~$ ls Unix/TP1
exo1
```

11. Renommer « *exo1* » en « *exercice1* ».

```
meriam@meriamVM:~$ mv Unix/TP1/exo1 Unix/TP1/exercice1
meriam@meriamVM:~$ ls Unix/TP1
exercice1
```

## B. Permissions et droits d'accès

1. Vérifier les droits d'accès du répertoire « *Unix* » et du fichier « *exercice1* ».

```
meriam@meriamVM:~$ ls -ld Unix/
drwxrwxr-x 4 meriam meriam 4096 sept. 21 16:34 Unix/

meriam@meriamVM:~$ ls -l Unix/TP1/exercice1
-rw-rw-r-- 1 meriam meriam 0 sept. 20 14:49 Unix/TP1/exercice1
```

2. Modifier les droits d'accès au fichier « *exercice1* » de telle sorte que tout le monde aura le droit de supprimer ce fichier.

```
meriam@meriamVM:~$ chmod a+w Unix/TP1/exercice1
meriam@meriamVM:~$ ls -l Unix/TP1/exercice1
-rw-rw-rw- 1 meriam meriam 0 sept. 20 14:49 Unix/TP1/exercice1
```

3. Interdire l'accès en lecture au répertoire « *TP1* » pour tout le monde.

```
meriam@meriamVM:~$ chmod a-r Unix/TP1
meriam@meriamVM:~$ ls -ld Unix/TP1
d-wx-wx--x 2 meriam meriam 4096 sept. 20 15:53 Unix/TP1
```

4. Remettre l'accès en lecture et interdire l'accès en lecture des fichiers qui sont dans « *TP1* ».

```
meriam@meriamVM:~$ chmod a+r-x Unix/TP1
meriam@meriamVM:~$ ls -ld Unix/TP1
drw-rw-r-- 2 meriam meriam 4096 sept. 20 15:53 Unix/TP1
meriam@meriamVM:~$ cat Unix/TP1/exercice1
cat: Unix/TP1/exercice1: Permission denied
```

## C. Liens physiques vs Liens symboliques

1. Afficher le numéro d'inode du fichier « *exercice1* ».

**NB : Il faut tout d'abord remettre l'accès en lecture des fichiers qui sont dans « *TP1* »**

```
meriam@meriamVM:~$ chmod a+x Unix/TP1/

meriam@meriamVM:~$ ls -li Unix/TP1
total 0
131154 -rw-rw-rw- 1 meriam meriam 0 sept. 20 14:49 exercice1
```

2. Créer un lien physique « *lp\_exe* » vers le fichier « *exercice1* » dans « *TP2* ».

```
meriam@meriamVM:~$ ln Unix/TP1/exercice1 Unix/TP2/lp_exe
meriam@meriamVM:~$ ls Unix/TP2/
lp_exe
```

3. Modifier le contenu du fichier « *lp\_exe* ». Qu'est-ce qu'on peut constater pour le fichier « *exercice1* » ? Réciproquement, modifier « *exercice1* », lire « *lp\_exe* ». Conclure.

```
meriam@meriamVM:~$ gedit Unix/TP2/lp_exe
meriam@meriamVM:~$ cat Unix/TP2/lp_exe
Bonjour tout le monde
meriam@meriamVM:~$ cat Unix/TP1/exercice1
Bonjour tout le monde
meriam@meriamVM:~$ gedit Unix/TP1/exercice1
meriam@meriamVM:~$ cat Unix/TP1/exercice1
Bonjour tout le monde
Hello everybody
meriam@meriamVM:~$ cat Unix/TP2/lp_exe
Bonjour tout le monde
Hello everybody
```

Un lien physique est une copie dynamique d'un fichier. Le fichier d'origine et le lien physique pointe vers le même contenu sur le disque dur,...

4. Créer un lien symbolique « *ls\_exe* » vers le fichier « *exercice1* » dans « *TP2* ».

```
meriam@meriamVM:~$ ln -s ~/Unix/TP1/exercice1 ~/Unix/TP2/ls_exe
meriam@meriamVM:~$ ls -l Unix/TP2/
total 4
-rw-rw-rw- 2 meriam meriam 38 sept. 21 14:38 lp_exe
lrwxrwxrwx 1 meriam meriam 31 sept. 21 15:49 ls_exe -> /home/meriam/Unix/TP1/exe
rcice1
meriam@meriamVM:~$ cat Unix/TP2/ls_exe
Bonjour tout le monde
Hello everybody
```

**NB :** Pour pouvoir accéder au lien symbolique après création (afficher son contenu), il faut lors de sa création spécifier son chemin absolu ainsi que le chemin absolu de son fichier d'origine.

5. Déplacer le fichier « *exercice1* » dans le répertoire « *Unix* ». Essayer à nouveau d'afficher les contenus de « *lp\_exe* » et « *ls\_exe* ». Supprimer le fichier « *exercice1* ». Qu'est-ce qu'on peut constater ?

```
meriam@meriamVM:~$ mv Unix/TP1/exercice1 Unix/
meriam@meriamVM:~$ ls Unix/
exercice1 TP1 TP2
meriam@meriamVM:~$ cat Unix/TP2/lp_exe
Bonjour tout le monde
Hello everybody
meriam@meriamVM:~$ cat Unix/TP2/ls_exe
cat: Unix/TP2/ls_exe: No such file or directory
```

```
meriam@meriamVM:~$ rm Unix/exercice1
meriam@meriamVM:~$ ls Unix/
TP1 TP2
meriam@meriamVM:~$ cat Unix/TP2/lp_exe
Bonjour tout le monde
Hello everybody
meriam@meriamVM:~$ cat Unix/TP2/ls_exe
cat: Unix/TP2/ls_exe: No such file or directory
```

On constate que le lien physique reste accessible alors que le lien symbolique ne l'est plus.

## D. Recherche des fichiers

1. Chercher dans le répertoire */etc* tous les fichiers dont l'extension est « **.conf** »

```
meriam@meriamVM:~$ find /etc -type f -name *.conf
/etc/apg.conf
/etc/ipp-usb/ipp-usb.conf
/etc/environment.d/90atk-adaptor.conf
/etc/environment.d/90qt-a11y.conf
/etc/gai.conf
/etc/debconf.conf
/etc/sysctl.conf
/etc/ubuntu-advantage/uaclient.conf
/etc/fprintd.conf
/etc/modules-load.d/cups-filters.conf
/etc/systemd/sleep.conf
/etc/systemd/resolved.conf
/etc/systemd/user.conf
/etc/systemd/logind.conf
/etc/systemd/networkd.conf
/etc/systemd/pstore.conf
/etc/systemd/system.conf
/etc/systemd/oomd.conf
/etc/systemd/timesyncd.conf
/etc/systemd/journald.conf
```

2. Chercher dans le répertoire */etc* tous les répertoires ayant les droits d'accès **rwX r-X r-X**

```
meriam@meriamVM:~$ find /etc -type d -perm 755
/etc
/etc/ipp-usb
/etc/ssh
/etc/ssh/ssh_config.d
/etc/kernel
/etc/kernel/postinst.d
/etc/kernel/install.d
/etc/kernel/postrm.d
/etc/kernel/preinst.d
/etc/netplan
/etc/pki
/etc/pki/fwupd-metadata
/etc/pki/fwupd
/etc/environment.d
/etc/ppp
/etc/ppp/ipv6-up.d
/etc/ppp/ipv6-down.d
```

3. Chercher dans le **répertoire personnel** tous les dossiers dont la taille est supérieure à **3Mo**

```
meriam@meriamVM:~$ find ~ -type d -size +3M
meriam@meriamVM:~$
```



- Chercher dans le répertoire */etc* tous les répertoires modifiés depuis une **semaine** et dont le nom contient **3 caractères**

```
meriam@meriamVM:~$ find /etc -type d -mtime 7 -name ???
find: '/etc/ssl/private': Permission denied
find: '/etc/polkit-1/localauthority': Permission denied
find: '/etc/cups/ssl': Permission denied
```

- Supprimer tous les fichiers du **répertoire personnel** dont le nom commence par **l** ou **L**.

```
meriam@meriamVM:~$ find ~ -type f -name [lL]*
/home/meriam/Unix/TP2/lp_exe
/home/meriam/.local/share/keyrings/login.keyring
/home/meriam/.cache/tracker3/files/locale-for-miner-apps.txt
/home/meriam/.cache/tracker3/files/last-crawl.txt
```

```
meriam@meriamVM:~$ find ~ -type f -name [lL]* -exec rm {} \;
meriam@meriamVM:~$ find ~ -type f -name [lL]*
meriam@meriamVM:~$
```

## E. Filtres

- Editer un nouveau fichier *exercice2* dans « *Unix* » en insérant le contenu suivant :

Module:	Enseignant:	Niveau
Système et Réseau:	Mohamed:	1
Administration Système Unix:	Salah:	3
Système et scripting:	Salah:	2
Techniques de Conteneurisation:	Mohamed:	4
Cloud et sécurité:	Fatma:	5
Fondements des réseaux:	Ahmed:	2
Initiation au cloud:	Fatma:	4
Mathématiques:	Hedi:	1
Switched networks:	Asma:	3
Livraison continue:	Amine:	5
IP Net Routing:	Asma:	3
Sécurité informatique:	Rim:	4
Réseaux de communication:	Ahmed:	2
Algorithmique:	Ali:	1
Programmation C:	Mohamed:	1

```
meriam@meriamVM:~$ gedit Unix/exercice2
```

- Afficher le contenu du fichier *exercice2*

```
meriam@meriamVM:~$ cat Unix/exercice2
Module:                                Enseignant:                Niveau
Système et Réseau:                    Mohamed:                    1
Administration Système Unix:          Salah:                      3
Système et scripting:                 Salah:                      2
Techniques de Conteneurisation:       Mohamed:                    4
Cloud et sécurité:                    Fatma:                     5
Fondements des réseaux:               Ahmed:                      2
Initiation au cloud:                  Fatma:                     4
Mathématiques:                        Hedi:                      1
Switched networks:                    Asma:                      3
Livraison continue:                   Amine:                      5
IP Net Routing:                       Asma:                      3
Sécurité informatique:                 Rim:                       4
Réseaux de communication:             Ahmed:                      2
Algorithmique:                        Ali:                       1
Programmation C:                      Mohamed:                    1
```

3. Afficher les noms des modules et les enseignants

```
meriam@meriamVM:~$ cut -d: -f 1,2 Unix/exercice2
Module:                                Enseignant
Système et Réseau:                    Mohamed
Administration Système Unix:          Salah
Système et scripting:                 Salah
Techniques de Conteneurisation:       Mohamed
Cloud et sécurité:                    Fatma
Fondements des réseaux:               Ahmed
Initiation au cloud:                  Fatma
Mathématiques:                        Hedi
Switched networks:                    Asma
Livraison continue:                   Amine
IP Net Routing:                       Asma
Sécurité informatique:                 Rim
Réseaux de communication:             Ahmed
Algorithmique:                        Ali
Programmation C:                      Mohamed
```

4. Afficher les modules dont le nom commence par S

```
meriam@meriamVM:~$ cut -d: -f 1 Unix/exercice2 | grep ^S
Système et Réseau
Système et scripting
Switched networks
Sécurité informatique
```

5. Donner le nombre de modules enseignés par Mr **Mohamed**

```
meriam@meriamVM:~$ grep Mohamed Unix/exercice2 | wc -l
3
```

6. Afficher la 6<sup>ème</sup> et 7<sup>ème</sup> ligne du fichier



```
meriam@meriamVM:~$ tail +2 Unix/exercice2 | tail +6 | head -2
Fondements des réseaux:      Ahmed:      2
Initiation au cloud:         Fatma:      4
```

7. Trier par ordre alphabétique les noms des modules

```
meriam@meriamVM:~$ tail +2 Unix/exercice2 | sort
Administration Système Unix:  Salah:      3
Algorithmique:               Ali:        1
Cloud et sécurité:           Fatma:      5
Fondements des réseaux:      Ahmed:      2
Initiation au cloud:         Fatma:      4
IP Net Routing:              Asma:       3
Livraison continue:          Amine:      5
Mathématiques:               Hedi:       1
Programmation C:             Mohamed:     1
Réseaux de communication:    Ahmed:      2
Sécurité informatique:        Rim:        4
Switched networks:           Asma:       3
Système et Réseau:           Mohamed:     1
Système et scripting:        Salah:      2
Techniques de Conteneurisation: Mohamed:     4
```

8. Trier par ordre croissant le fichier *exercice1* selon le niveau et afficher le résultat dans un nouveau fichier « **Rapport** »

```
meriam@meriamVM:~$ tail +2 Unix/exercice2 | sort -t: -k3n > Rapport
meriam@meriamVM:~$ cat Rapport
Algorithmique:               Ali:        1
Mathématiques:               Hedi:       1
Programmation C:             Mohamed:     1
Système et Réseau:           Mohamed:     1
Fondements des réseaux:      Ahmed:      2
Réseaux de communication:    Ahmed:      2
Système et scripting:        Salah:      2
Administration Système Unix:  Salah:      3
IP Net Routing:              Asma:       3
Switched networks:           Asma:       3
Initiation au cloud:         Fatma:      4
Sécurité informatique:        Rim:        4
Techniques de Conteneurisation: Mohamed:     4
Cloud et sécurité:           Fatma:      5
Livraison continue:          Amine:      5
```

9. Afficher les noms des modules des 1<sup>ère</sup> et 2<sup>ème</sup> années dans le fichier « **Rapport** »

```
meriam@meriamVM:~$ grep [12] Unix/exercice2 | cut -d: -f 1 >> Rapport
```

```
meriam@meriamVM:~$ cat Rapport
Algorithmique: Ali: 1
Mathématiques: Hedi: 1
Programmation C: Mohamed: 1
Système et Réseau: Mohamed: 1
Fondements des réseaux: Ahmed: 2
Réseaux de communication: Ahmed: 2
Système et scripting: Salah: 2
Administration Système Unix: Salah: 3
IP Net Routing: Asma: 3
Switched networks: Asma: 3
Initiation au cloud: Fatma: 4
Sécurité informatique: Rim: 4
Techniques de Conteneurisation: Mohamed: 4
Cloud et sécurité: Fatma: 5
Livraison continue: Amine: 5
Système et Réseau
Système et scripting
Fondements des réseaux
Mathématiques
Réseaux de communication
Algorithmique
Programmation C
```